

Tema 12 - El Procesador MicroBlaze

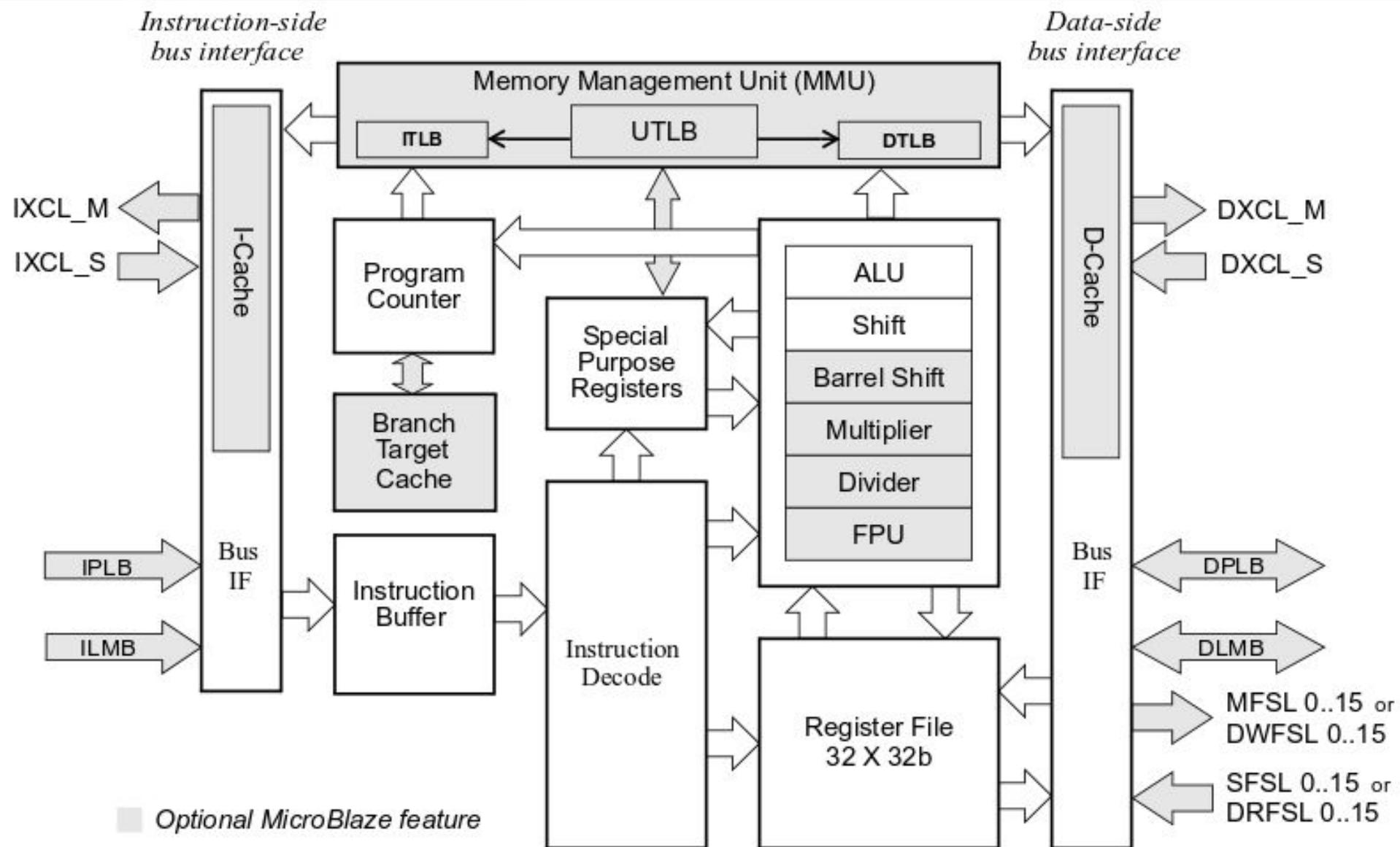
Sistemas Electrónicos para Automatización
Grado en Ingeniería Electrónica, Robótica y
Mecatrónica

Hipólito Guzmán Miranda

Contenido

- Arquitectura
- Buses
- Configuración
- El software EDK (Embedded Development Kit)
- Periféricos personalizados por el usuario

Arquitectura



Siglas

- FPU: Floating Point Unit
- I-* : Instruction *, D-*: Data *
- PLB: Processor Local Bus
- LMB: Local Memory Bus
- TLB: Transaction Look-aside Buffer

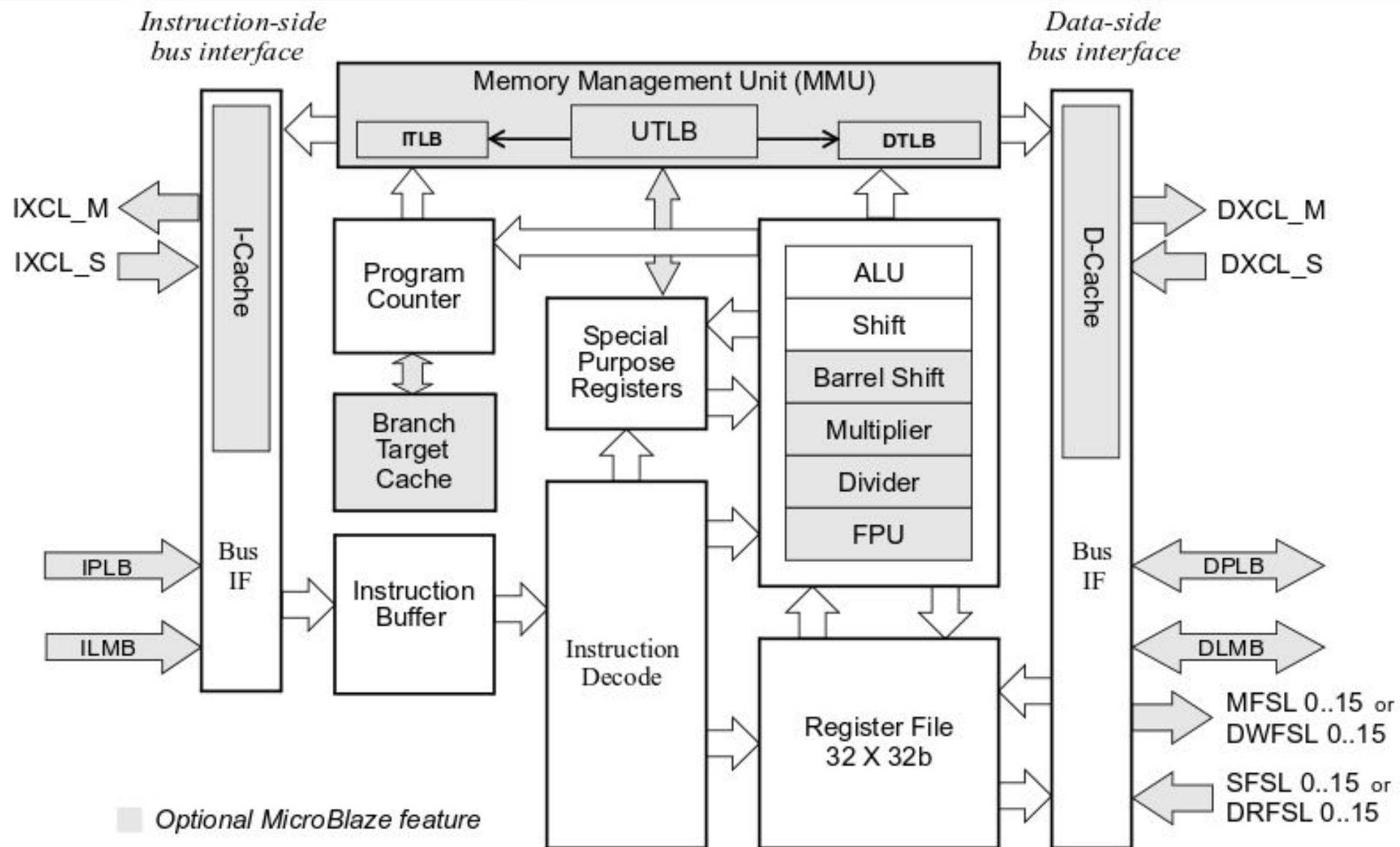
Siglas (II)

- **FSL: Fast Simplex Link**
 - (M: Master, S: Slave, DW: Direct connection - Write, DR: Direct Connection - Read)
- **XCL: Xilinx CacheLink interface**
 - (M: Master, S: Slave)
- **IF: Interface**

Arquitectura

- Ya que es un soft processor, su arquitectura es configurable
- Bloques básicos y bloques opcionales
- Cachés, MMU, FPU, Divisor, opcionales
- El 'coste' es en recursos ocupados en la FPGA

Arquitectura



Buses

- Buses de Datos separados de buses de Instrucciones
- XCL : para comunicación eficiente con las cachés de datos e instrucciones
- LMB : para acceder a las memorias de datos e instrucciones (BRAMs)
- FSL : conexión rápida y sencilla a coprocesador

Buses para periféricos

Dos opciones:

- PLB (Processor Local Bus).
- AXI (Advanced eXtensible Interface).

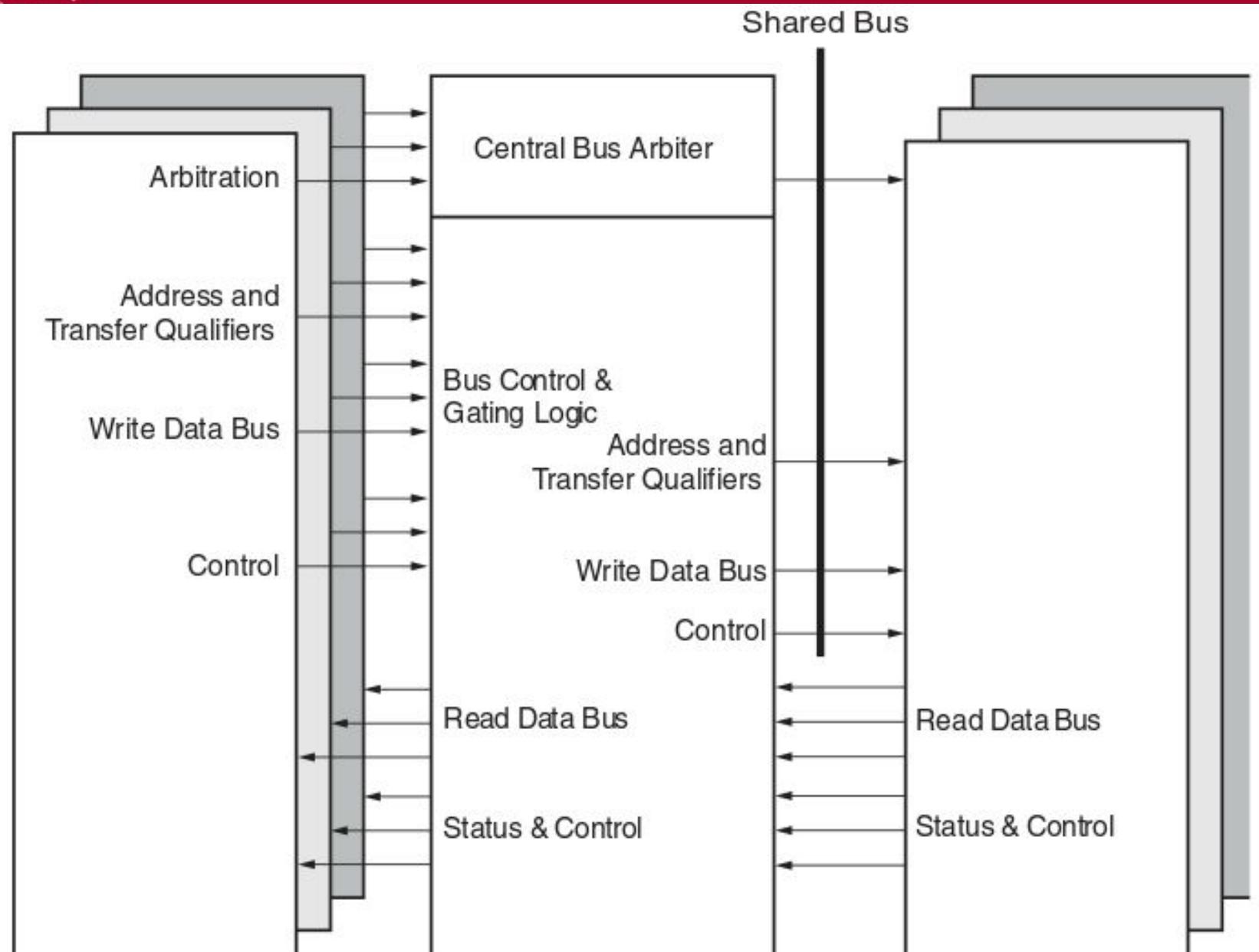
Soporte a partir de Spartan-6. Versiones ISE/EDK de la 13.1 en adelante.

Nosotros tendremos que usar PLB (ya que ISE 14.7 no cabe en los PC del CDC)

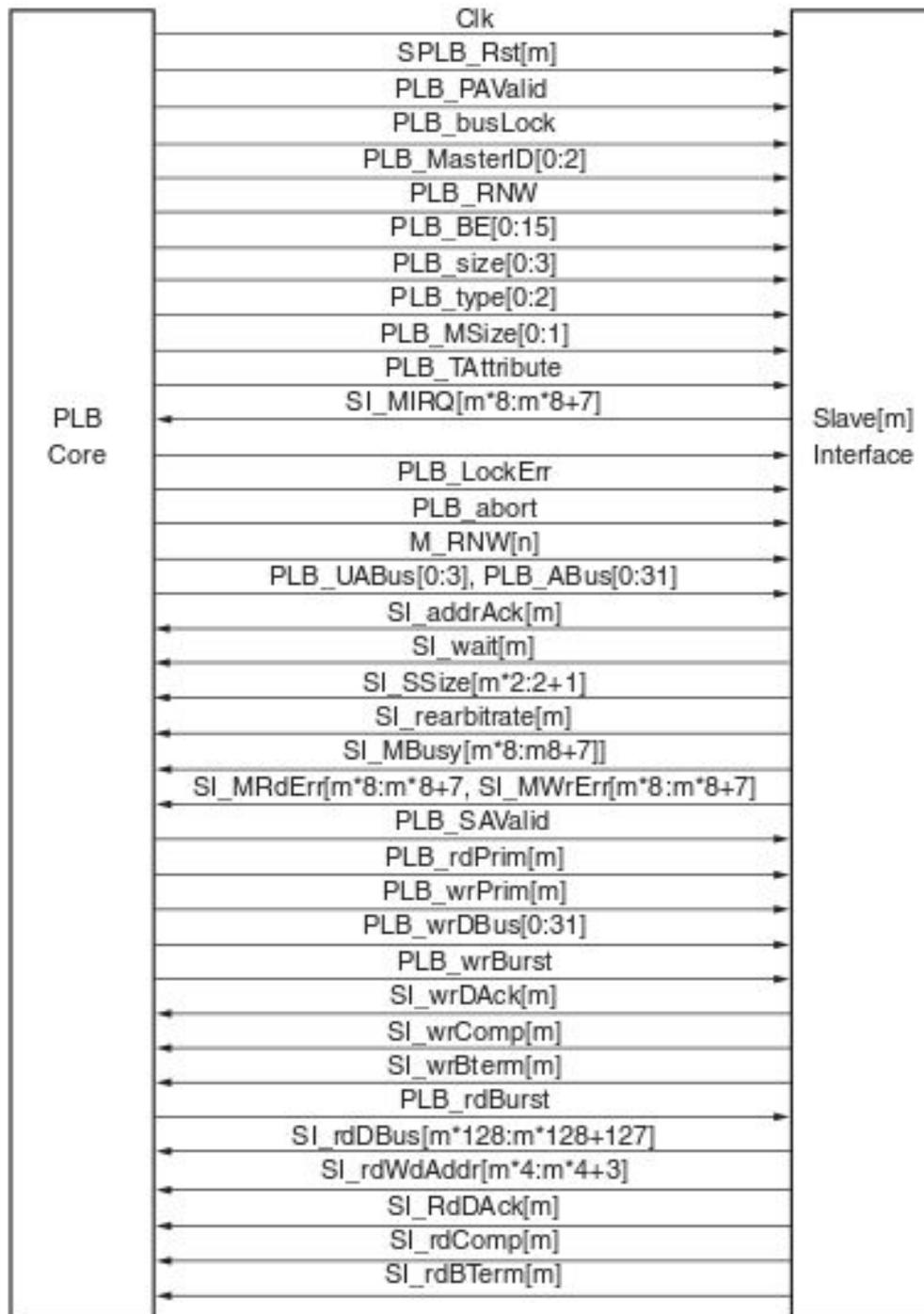
PLB (Processor Local Bus)

- PLB es una especificación de bus
- Se usa PLB para periféricos que requieren mayores prestaciones en la comunicación
- En algunos casos se usa OPB (On-Chip Peripheral Bus) para dispositivos más lentos

PLB



P



Bus)

AXI (Advanced eXtensible Interface)

- AXI es una especificación de interconexión
- Mayores anchos de banda
- Es parte de AMBA (Advanced Microprocessor Bus Architecture), junto con AHB (Advanced High-performance Bus) y APB (Advanced Peripheral Bus)

El Software EDK

- EDK = Embedded Development Kit
- Permite implementar el soft-core MicroBlaze
- (También permite utilizar los PPC440 que están integrados en Silicio en algunas FPGA de gama alta como las Virtex5 serie FX)

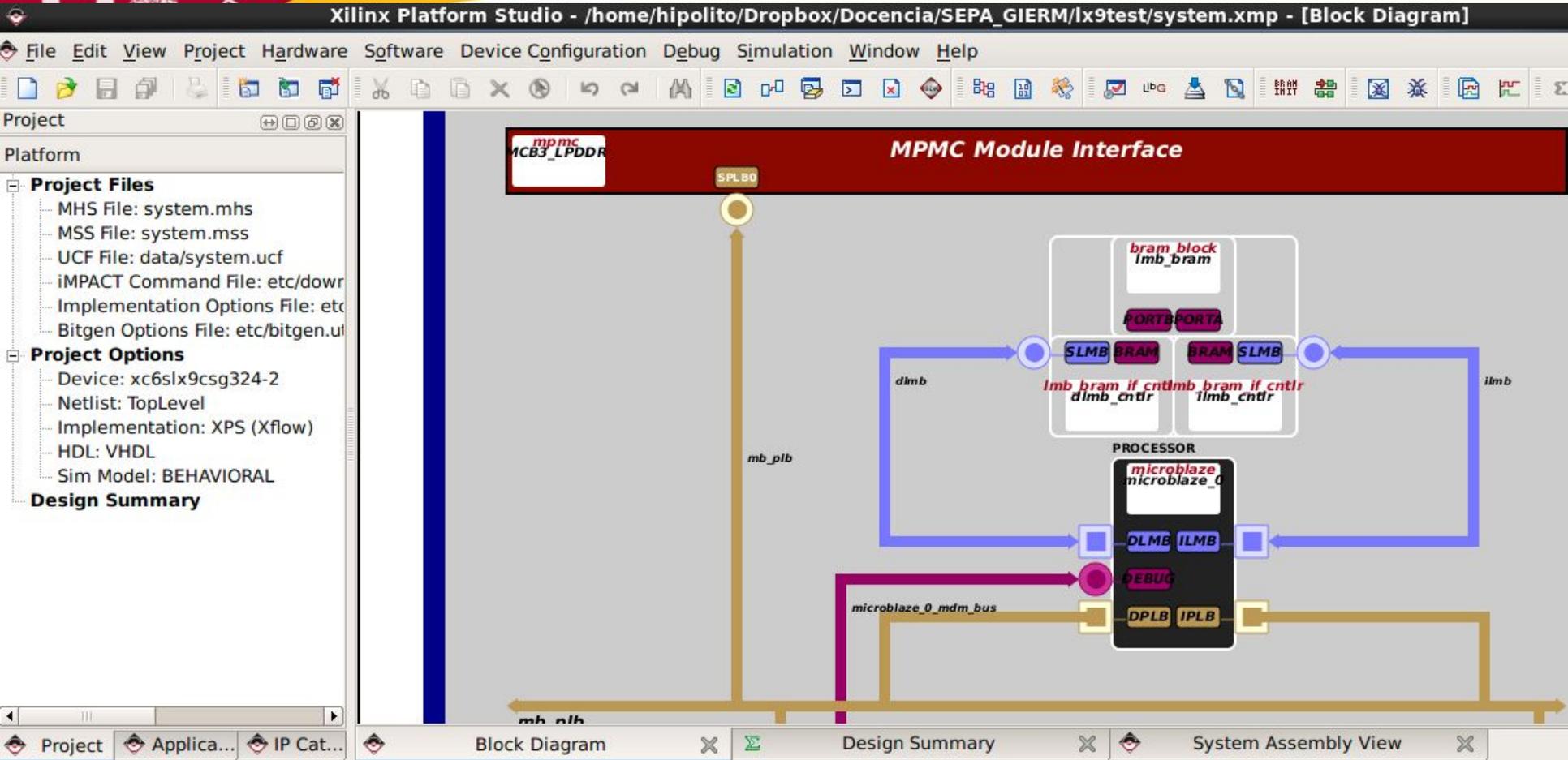


Diagrama de bloques



El Software EDK

Xilinx Platform Studio - /home/hipolito/Dropbox/Docencia/SEPA_GIERM/lx9test/system.xmp - [System Assembly View]

File Edit View Project Hardware Software Device Configuration Debug Simulation Window Help

Project Platform

Project Files

- MHS File: system.mhs
- MSS File: system.mss
- UCF File: data/system.ucf
- iMPACT Command File: etc/dowr
- Implementation Options File: etc
- Bitgen Options File: etc/bitgen.u

Project Options

- Device: xc6slx9csg324-2
- Netlist: TopLevel
- Implementation: XPS (Xflow)
- HDL: VHDL
- Sim Model: BEHAVIORAL

Design Summary

Legend

- Master Slave Master/Slave Target Initiator Connected Unconnected
- Production License (paid) License (eval) Local Pre Production Beta Development
- Superseded Discontinued

Block Diagram Design Summary System Assembly View

Bus Interfaces

Name	Bus Name	IP Type	IP Version
microblaze_0		microblaze	7.30.a
DLMB	dlmb		
ILMB	ilmb		
DPLB	mb_plb		
IPLB	mb_plb		
DXCL	microblaze_...		
IXCL	microblaze_...		
DEBUG	microblaz...		
TRACE	microblaze_...		
lmb_bram		★ bram_bl...	1.00.a
dlmb_cntlr		★ lmb_bra...	2.10.b
ilmb_cntlr		★ lmb_bra...	2.10.b
MCB3_LPDDR		★ mpmc	6.00.a
mdm_0		★ mdm	1.00.g
SPLB	mb_plb		
MFSLO	No Conne...		
MBDEBUG_0	microblaze_...		
XMTC	mdm_0_XMTC		
Ethernet MAC		★ xns eth...	4.00.a

Periféricos y conexiones a los buses



El Software EDK

Xilinx Platform Studio - /home/hipolito/Dropbox/Docencia/SEPA_GIERM/lx9test/system.xmp - [System Assembly View]

File Edit View Project Hardware Software Device Configuration Debug Simulation Window Help

Project Platform

Project Files

- MHS File: system.mhs
- MSS File: system.mss
- UCF File: data/system.ucf
- iMPACT Command File: etc/down
- Implementation Options File: etc
- Bitgen Options File: etc/bitgen.ut

Project Options

- Device: xc6slx9csg324-2
- Netlist: TopLevel
- Implementation: XPS (Xflow)
- HDL: VHDL
- Sim Model: BEHAVIORAL

Design Summary

Bus Interfaces Ports **Addresses**

Instance	Base Name	Base Address	High Address	Size	Bus Interface(s)	Bus Name	Lock
microblaze_0's Address ...							
dlmb_cntlr	C_BASEADDR	0x00000000	0x00001FFF	8K	SLMB	dlmb	<input type="checkbox"/>
ilmb_cntlr	C_BASEADDR	0x00000000	0x00001FFF	8K	SLMB	ilmb	<input type="checkbox"/>
Ethernet_MAC	C_BASEADDR	0x81000000	0x8100FFFF	64K	SPLB	mb_plb	<input type="checkbox"/>
LEDs_4Bits	C_BASEADDR	0x81400000	0x8140FFFF	64K	SPLB	mb_plb	<input type="checkbox"/>
DIP_Switch_4Bits	C_BASEADDR	0x81420000	0x8142FFFF	64K	SPLB	mb_plb	<input type="checkbox"/>
CDCE913_I2C	C_BASEADDR	0x81600000	0x8160FFFF	64K	SPLB	mb_plb	<input type="checkbox"/>
SPI_FLASH	C_BASEADDR	0x83400000	0x8340FFFF	64K	SPLB	mb_plb	<input type="checkbox"/>
USB_UART	C_BASEADDR	0x84000000	0x8400FFFF	64K	SPLB	mb_plb	<input type="checkbox"/>
mdm_0	C_BASEADDR	0x84400000	0x8440FFFF	64K	SPLB	mb_plb	<input type="checkbox"/>
MCB3_LPDDR	C_MPMC_BA...	0x8C000000	0x8FFFFFFF	64M	:SPLB0		<input type="checkbox"/>

Legend

- Master Slave Master/Slave Target Initiator Connected Unconnected
- Production License (paid) License (eval) Local Pre Production Beta Development
- Superseded Discontinued

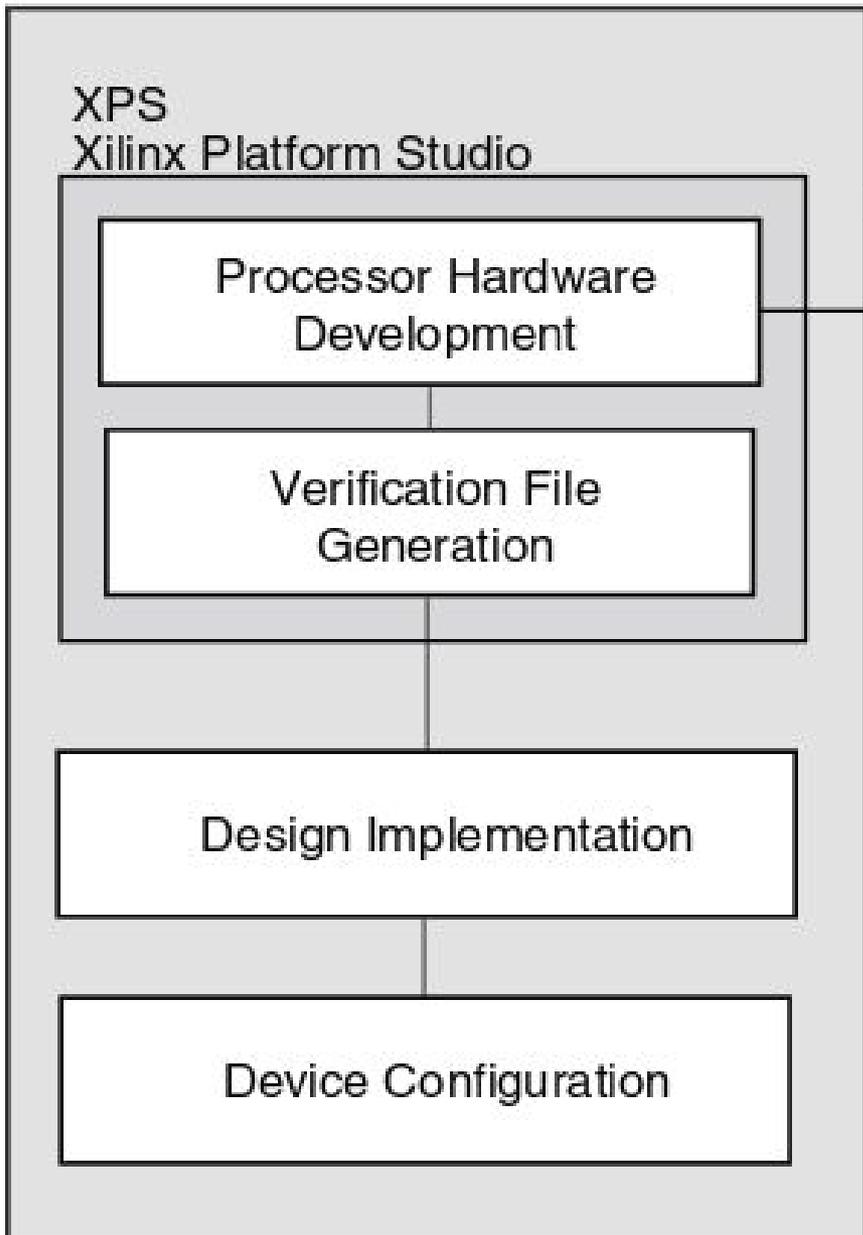
Project Applica... IP Cat... Block Diagram Design Summary System Assembly View

Mapa de direcciones

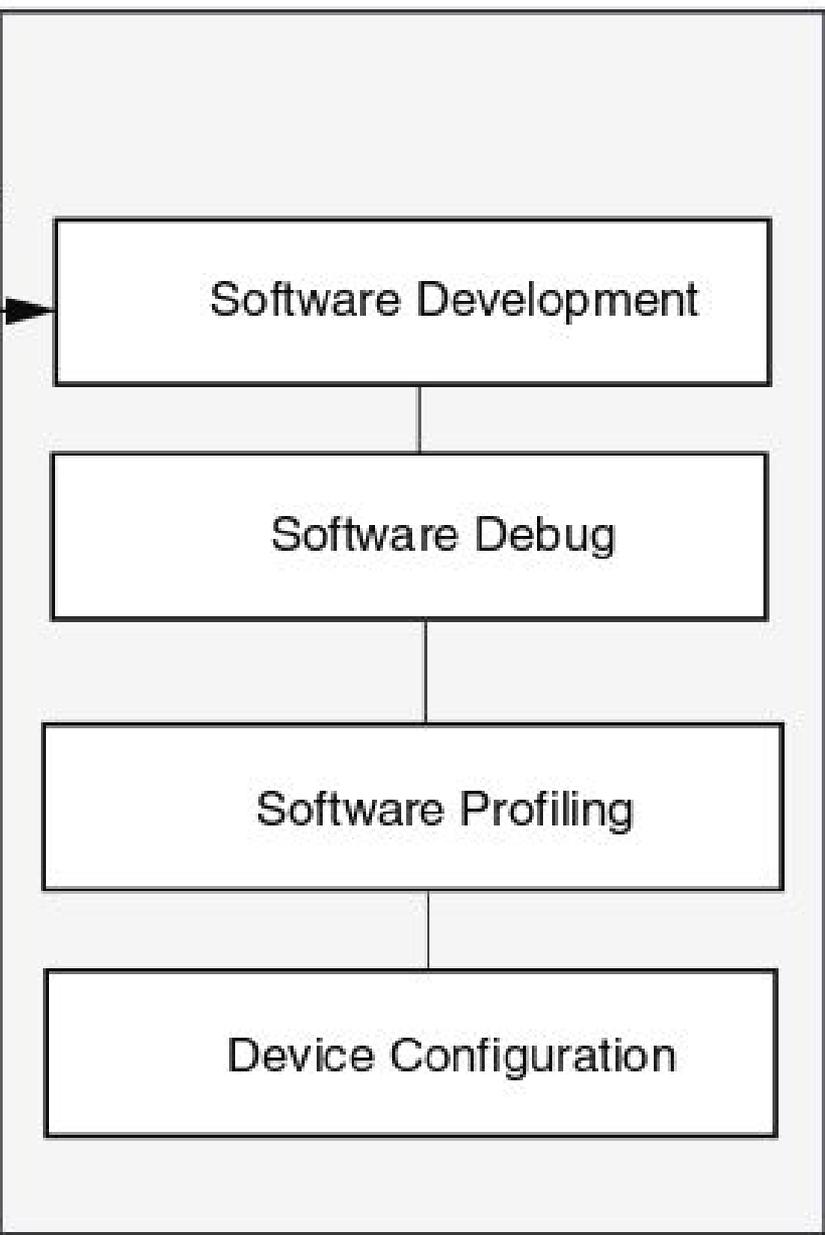
El Software EDK: Flujo de diseño

- Crear microprocesador
- Configurar periféricos
- Proyecto software
- Librerías
- Síntesis
- Implementación
- Compilación del software
- Depuración del software
- Generación del bitstream con programa

ISE Integrated Software Environment



SDK Software Development Kit



Hardware Platform



Crear Microprocesador

- Se configura desde cero o a partir de un fichero .XBD (Xilinx Board Definition) que trae un sistema básico
- Es recomendable utilizar un XBD, si existe para la tarjeta de desarrollo que estemos utilizando



Base System Builder

Welcome Board System **Processor** Peripheral Cache Application Summary

Processor Configuration

Configure the processor(s).

Reference Clock Frequency MHz

Processor 1 Configuration

Processor Type

System Clock Frequency MHz

Local Memory

Debug Interface

Enable Floating Point Unit

Crear Microprocesador

- Una vez creado el proyecto se puede (re)configurar el procesador:

MicroBlaze™

Configuration Wizard

Select configuration:

-  Current Settings
-  Minimum Area
-  Maximum Performance
-  Maximum Frequency
-  Linux with MMU
-  Low-end Linux with MMU

Welcome to MicroBlaze Configuration Wizard

- Select a predefined configuration in the list to the left. Information about the selected configuration is shown below. *Each predefined configuration completely changes the MicroBlaze parameters.*
- To modify the configuration, click on the *Next* button, click on *Advanced* to directly access parameters in a tabbed interface, or click on *OK* to accept the configuration and close the dialog.

- Select implementation to optimize area (with lower instruction throughput)
- Enable Debug
- Use Instruction and Data Caches
- Enable Exceptions
- Use Memory Management

[Advanced](#)[Next >](#)

Frequency



Area



Performance



BRAM

71

DSP48A1

10

Changed MicroBlaze settings according to template "Maximum Performance"

Page 2 of 4 - Performance and Instructions

Optimization

Enable Branch Target Cache

Branch Target Cache Size

DEFAULT

Instructions

Enable Barrel Shifter

Enable Floating Point Unit

NONE

Enable Integer Multiplier

MUL32

Enable Integer Divider Enable Additional Machine Status Register Instructions Enable Pattern Comparator

Advanced

< Back

Next >

Frequency

Area

Performance

BRAM

DSP48A1

Configurar Periféricos

- Se configuran los periféricos iniciales al crear el proyecto:



Peripheral Configuration

To add a peripheral, drag it from the "Available Peripherals" to the processor peripheral list. To change a core parameter, click on the peripheral.

Available Peripherals

Peripheral Names

- IO Devices
- Internal Peripherals
 - xps_bram_if_cntlr
 - xps_timebase_wdt
 - xps_timer**

Add >

< Remove

Processor 1 (MicroBlaze) Peripherals

Select All

Core	Parameter
Ethernet_MAC	
Core	xps_ethernetlite
Use Interrupt	<input type="checkbox"/>
FUEL_GAUGE	
Core: xps_iic	
LEDs_4Bits	
Core: xps_gpio	
MCB3_LPDDR	
Core: mpmc, Address Width: 13,...	
Push_Buttons_3Bits	
Core: xps_gpio	
SPI_FLASH	
Core: xps_spi, C_NUM_SS_BITS: ...	
USB_UART	
Core: xps_uartlite, Baud Rate: 9...	
dlmb_cntlr	
Core: lmb_bram_if_cntlr	
ilmb_cntlr	
Core: lmb_bram_if_cntlr	

More Info

< Back

Next >

Cancel

Configurar Periféricos

- Y se pueden añadir posteriormente utilizando el 'IP Catalog':



El Procesador MicroBlaze

Xilinx Platform Studio - /home/hipolito/Dropbox/Docencia/SEPA_GIERM/zz/system.xmp - [System Assembly View]

File Edit View Project Hardware Software Device Configuration Debug Simulation Window Help

IP Catalog

Description IP Versi

- EDK Install
 - Analog
 - Bus and Bridge
 - Clock, Reset and Interrupt
 - Communication High-Speed
 - Communication Low-Speed
 - DMA and Timer
 - Debug
 - General Purpose IO
 - XPS General Purpose IO 2.00.a
 - IO Modules
 - Interprocessor Communication
 - Memory and Memory Controller
 - PCI
 - Peripheral Controller
 - Processor
 - Utility
- Global Peripheral Repository0
- Project Local pcores
 - USER

Bus Interfaces Ports Addresses

Name	Bus Name	IP Type	IP Version
dmb		★ lmb_v10	1.00.a
ilmb		★ lmb_v10	1.00.a
mb_plb		★ plb_v46	1.04.a
+ microblaze_0		★ microblaze	7.30.a
lmb_bram		★ bram_bl...	1.00.a
+ dmb_cntlr		★ lmb_bra...	2.10.b
+ ilmb_cntlr		★ lmb_bra...	2.10.b
+ MCB3_LPDDR		★ mpmc	6.00.a
+ mdm_0		★ mdm	1.00.g
+ Ethernet_MAC		★ xps_eth...	4.00.a
4Bits		★ xps_gpio	2.00.a
Buttons_3Bits		★ xps_gpio	2.00.a
GAUGE		★ xps_iic	2.03.a
LASH		★ xps_spi	2.01.b
UART		★ xps_uart...	1.01.a
_generator_0		★ clock_ge...	4.00.a
sys_reset_0		★ proc_sys...	2.00.a

Bus Interface Filters

- By Connection
 - Connected
 - Unconnected
- By Bus Standard
 - LMB
 - PLBV46
 - FSL
 - Xilinx Point To Point
- By Interface Type
 - Slaves
 - Masters
 - Master Slaves
 - Monitors
 - Targets
 - Initiators

Legend

- Master
- Slave
- Master/Slave
- Target
- Initiator
- Connected
- Unconnected
- ★ Production
- License (paid)
- License (eval)
- Local
- Pre Production
- Beta
- Development
- Superseded
- Discontinued

Project Applications IP Catalog Block Diagram Design Summary System Assembly View

Console

```
/home/hipolito/opt/Xilinx/12.1/ISE_DS/EDK/data/xml/xslscripts/ConvertEdwardVersion.xsl
Generated Block Diagram SVG
```

Console Warnings Errors

Proyecto Software

- Por defecto se generan dos proyectos de software:



Base System Builder

Welcome Board System Processor Peripheral Cache **Application** Summary

Application Configuration

Configure the example applications.

Example Applications

Application	Option Value
[-] Test microblaze_0	
[-] Standard IO	USB_UART
[-] Boot Memory	ilmb_cntlr
[-] Memory Test	TestApp_Memory_microblaze_0
[-] Instructions	ilmb_cntlr
[-] Data	dlmb_cntlr
[-] Peripheral Test	TestApp_Peripheral_microblaze_0
[-] Instructions	MCB3_LPDDR
[-] Data	MCB3_LPDDR
[-] Interrupt Vector	ilmb_cntlr

[More Info](#) < Back Next > Cancel

Proyecto Software

- Estos proyectos se pueden editar una vez generado el diseño:

(Además, doble click en los ficheros `.{c,h}` para editar el código en EDK)

Applications

Software Projects

- [-] Add Software Application Project...
- [-] Default: microblaze_0_bootloop
- [-] Default: microblaze_0_xmdstub
- [-] **Project: TestApp_Memory_microblaze_0**
 - [+] Processor: microblaze_0
 - Executable: /home/hipolito/Dropbox/Docencia/SEPA_C
 - [+] Compiler Options
 - [+] Sources
 - [+] Headers
- [-] **Project: TestApp_Peripheral_microblaze_0**
 - [+] Processor: microblaze_0
 - Executable: /home/hipolito/Dropbox/Docencia/SEPA_C
 - [+] Compiler Options
 - [+] Sources
 - [+] Headers

Project Applications IP Catalog

Síntesis e Implementación

- Hardware -> Generate Netlist
- Hardware -> Generate Bitstream

Sólo con este bitstream el micro no arrancará, ya que no tiene ningún software en la memoria de instrucciones

Compilación del software

- Software -> Generate Libraries and BSPs (Board Support Packages)
- Software -> Build All User Applications
 - (O click derecho sobre una aplicación -> Build Project)

Generación del bitstream con programa

- Device Configuration -> Update Bitstream

Se parte de:

- Bitstream con memorias BRAM vacías
- Programa compilado

Se obtiene:

- Bitstream con memorias inicializadas

Ficheros de Configuración

- `<projname>.mhs` (Microprocessor Hardware Specification)
 - Puertos, Periféricos, rangos de direcciones, instancias
- `<projname>.mss` (Microprocessor Software Specification)
 - Drivers para los periféricos

Ficheros de Configuración

- `<projname>.xmp`
 - Proyecto de EDK. Entre otras cosas contiene:
FPGA utilizada, localización del .UCF,
localización de los sources y headers de las
aplicaciones software, script de linkado del
software.

Periféricos personalizados por el usuario

Principalmente dos opciones:

- **Conectar nuestro periférico a un GPIO**
 - Menores prestaciones
 - Más sencillo y rápido de prototipar
- **Conectar nuestro periférico al PLB**
 - Mayores prestaciones
 - Mayor dificultad

Periférico GPIO

Lo más sencillo es:

- Añadir un GPIO nuevo al MicroBlaze
- Declarar como External Ports las entradas/salidas apropiadas del GPIO
- Crear un proyecto .xise
- Añadir el .xps (proyecto EDK) con el MicroBlaze como 'source' al proyecto .xise
- Implementar nuestro periférico en VHDL con ISE

Periférico PLB

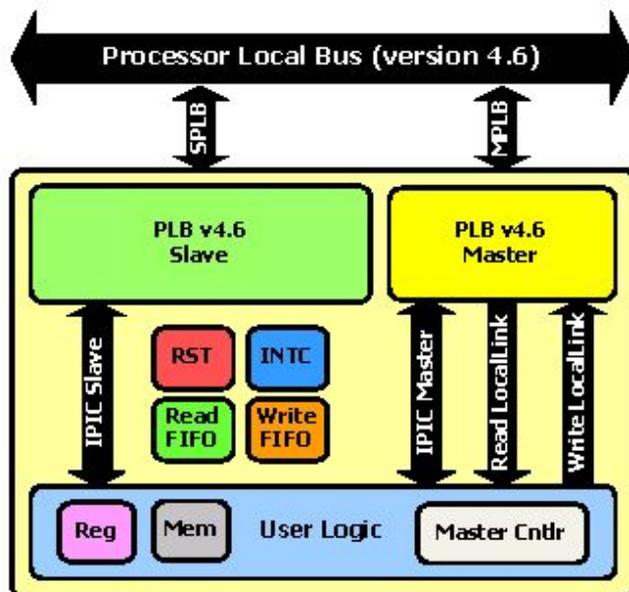
- El 'Create and Import peripheral wizard' de Xilinx nos genera una plantilla con la conexión al bus

IPIF (IP Interface) Services

Indicate the IPIF services required by your peripheral.



Your peripheral will be connected to the PLB (v4.6) interconnect through corresponding PLB IP Interface (IPIF) modules, which provide you with a quick way to implement the interface between the PLB interconnect and the user logic. Besides the standard functions like address decoding provided by the slave IPIF module, the wizard tool also offers other commonly used services and configurations to simplify the implementation of the design.



Slave service and configuration

Typically required by most peripherals for operations like logic control, status report, data buffering, multiple memory/address space access, and etc. (PLB slave interface will always be included).

- | | |
|--|--|
| <input type="checkbox"/> Software reset | <input checked="" type="checkbox"/> User logic software register |
| <input type="checkbox"/> Read/Write FIFO | <input type="checkbox"/> User logic memory space |
| <input type="checkbox"/> Interrupt control | <input checked="" type="checkbox"/> Include data phase timer |

Master service and configuration

Typically required by complex peripherals like Ethernet and PCI for commanding data transfers between regions (PLB master interface will be included if master service selected).

- User logic master

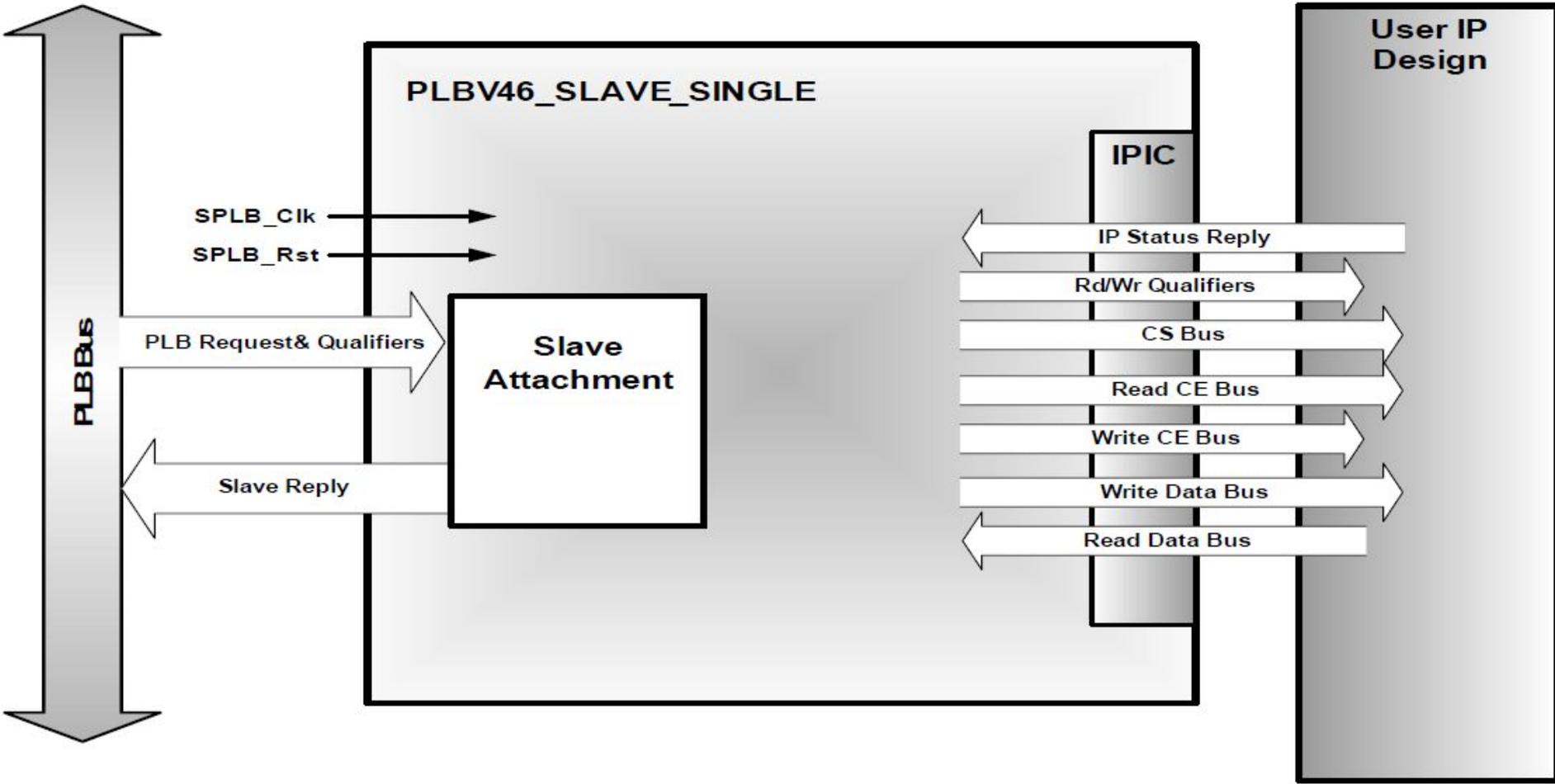
More Info

< Back

Next >

Cancel

Periféricos personalizados



Opciones para el interfaz PLB:

- FIFOs
- Interrupciones
- Registro accesible por software
- Espacio de memoria de usuario...

Lo que usemos dependerá de nuestras necesidades

Importante mirar bien la documentación!
(Incluyendo el botón de “More Info”)

¿Por dónde empezamos a describir?

- El asistente genera los ficheros de la plantilla en
`<project_dir>/pcores/<periph_name>`
- Leer bien el fichero
`<project_dir>/pcores/<periph_name>/dev1/README.txt`

¿Por dónde empezamos a describir? (II)

Dentro de `<project_dir>/pcores/<periph_name>` :

- `./data/<periph_name>.mpd`
(Microprocessor Peripheral Definition, aquí añadiremos puertos adicionales si los necesitamos)
- `./hdl/vhdl/<periph_name>.vhd` (top level)
- `./hdl/vhdl/user_logic.vhd` (lógica de usuario -> aquí describimos el funcionamiento)

Conclusiones

- Se pueden crear System-on-Chip complejos en FPGAs, con relativa facilidad, utilizando MicroBlaze
- La enorme configurabilidad del sistema nos obliga a mirar despacio la documentación
- También podemos desarrollar nuestros propios periféricos
- Vayamos poco a poco e intentemos que el microprocesador arranque ;)

Referencias

- [MicroBlaze Processor Reference Guide](#)
(UG081, v11.0 - EDK 12.1)
- [EDK Concepts, Tools, and Techniques: A Hands-On Guide to Effective Embedded System Design](#) (UG683 - EDK 12.1)
- [Logicore IP Processor Local Bus \(PLB\) Product Specification](#)